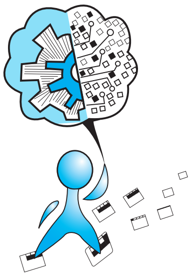


Hybrid Similarity Metrics for Neighbor Based Collaborative Filtering Performance Improvement

Mouneswari Pentakota

Pursuing Master of Technology in the stream of Data Science at Jawaharlal Nehru and Technology University, Guarajada, Vizianagram. Email: mounieswari123@gmail.com

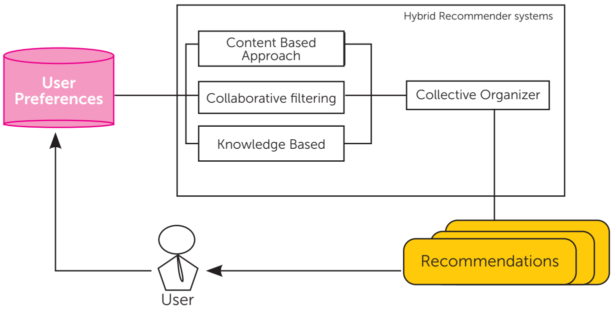
Keywords: Collaborative filtering, K-nearest-neighbor, Recommendation system, Item-based, Memory-based, Similarity measurement

Recommendation systems are a type of information filtering system that uses historical data to forecast a user's opinion or preference on a topic or item, allowing them to receive customised recommendations. They're common on e-commerce sites like amazon.com, online movie streaming services like Netflix, and social media sites like Facebook. With such a great number and variety of products, a recommendation system could also assist streaming services or online sellers in providing customers with recommendations according to their preferences.

This could improve the user experience when looking for products or services, leading to increased purchases, movie viewing, or service subscriptions. For example, statistics collected over three weeks in the summer of 2001 revealed that between 20% and 40% of Amazon purchases are attributable to recommended things that are not among the shop's 100,000 most popular items, while 60% of Netflix movies are chosen based on personalised recommendations. Furthermore, a recommendation system could produce additional money by introducing shoppers to new categories, in addition to greater direct revenue. As a result, a recommendation system might have a big impact on a company's revenue.

Note that while a 1% improvement in average MAE(Mean Absolute Error) and RMSE(Root Mean Squared Error) may seem insignificant, it can make a big difference in a user's ranking of the "top10" most suggested movies. As previously stated, recommendation systems can take one of three approaches: content-based filtering, collaborative filtering, or a combination of the two.

Hybrid Similarity metrics is implied to overcome the constraints of their individual approaches and increase performance such as prediction accuracy and scalability, this strategy combines two or more techniques or uses other techniques such as deep learning or clustering. However,it adds to the computational complexity, requiring more time and resources. Deep neural networks have had a lot of success in computer vision and natural language processing in recent years. Graph Convolutional Networks (GCNs) have also been successfully applied to recommendation systems.



Proposed Approach is to quantify the similarity between items, we suggest a new similarity measurement. Unlike standard CFs, which rely solely on rating-based similarity measures like adjusted cosine, Pearson correlation coefficient, or structural similarity measurement, we propose combining the two. Measures of resemblance based on ratings. The MovieLens and Netflix datasets are selected for comparison because they are the most extensively referenced in the literature. We evaluate and contrast two benchmarks for evaluating prediction accuracy: MAE (Mean Absolute Error) and RMSE (Root Mean Squared Error), and present experimental results on three widely used datasets: Netflix, MovieLens 100K, and MovieLens1M

On the MovieLens 100 K, 1 M, and Netflix datasets, our technique beats state-of-the-art collaborative filterings in terms of lower MAE with 1/3 to 1/2 the number of neighbours as compared to standard memory-based CFs. On the MovieLens 1 M dataset, memory-based CF with the suggested similarity measurement uses 1/2 to 1/39 wall time compared to state-ofthe-art model-based CFsFor non-cold start users, our method yields a 3% lower MAE and RMSE than traditional memory-based CFs on the MovieLens 100 K dataset.

Characteristics and Challenges are We concentrate on memory-based collaborative filtering since it may be used to any relational data without understanding what it contains. However, collaborative filtering faces several fundamental obstacles in predicting an accurate rating in real time. Sparsity Cold Start Problems are one of the most common problems that data sparsity can cause. Because collaborative filtering systems produce suggestions based on previous preferences, new users must rate a significant number of items for the recommendation system to learn their preferences and make valid recommendations. When users have just evaluated a few things, collaborative filtering systems are often unable to produce accurate recommendations.

Scalability Traditional collaborative filtering approaches will have scalability issues as the number of people and objects grows very large. Model-based techniques would find it difficult to react in real time to fresh user ratings in order to generate an updated suggestion because they train on the complete dataset. however, most users have only reviewed a small percentage of the total number of products, and memory-based techniques can react to new ratings and make predictions in real time, even for massive datasets. While using dimensionality reduction techniques like SVD to minimise scalability issues, model-based approaches suffer from computationally expensive matrix factorization and may lose essential information in the process

Curse of Dimensionality In order to find related objects or users, collaborative filtering must calculate similarities between them. Because there are many users and things, the pairwise similarities are estimated in high dimensions. The Hughes Phenomenon states that as dimensionality rises, the predictive power of a certain size of training samples decreases (Hughes, 1968).When all users or objects have the same similarities, memory-based CF is unable to locate the most comparable items or users, and so cannot provide a trustworthy forecast. The quality of hubness In the nearest neighbour lists of other items, certain items appear more frequently than others. Those things are typically popular high-rated items that do not contribute any personal preference information to suggestions because they may be enjoyed by a large number of users. They can act as noise, preventing memory-based CF from generating precise predictions.

Improvements over using rating or structural similarity alone

Compensate unpopular but similar items: We do not penalise similarities between items because of their unpopularity if both items have been evaluated by the majority of users. Instead of utilising a set global shrinkage factor to punish pairs of items with a small number of co-rated users, the structural similarity measurement compensates unpopular but heavily co-rated items by applying a local ratio of |Cij| to the local possible co-rated users |ii| and |ij|. We estimate the prediction accuracy to be higher than if we only used rating-based similarity measurement

Reduce Hubness Big hubs are those generally liked popular goods when calculated based on rating-based similarity measurement alone. Those goods don't add much to personal preferences, yet they're frequently mentioned as neighbours. The hubness of employing local ratio structural-based similarity weighted rating-based similarity should be lower than rating-based similarity alone. When just structural data is used to calculate similarities, huge hubs form when one user reviews unpopular goods that are rarely rated by other users. As a result, structural similarity will be high.

Full Picture We can anticipate whether a user likes or dislikes an item based on his or her opinions on highly connected neighbour items by merging the two. The strongly connected items are found using structural similarity assessment, and the user loves the item is determined using rating-based similarity measurement.

References :

[1] Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering, 17, 734–749. Aggarwal, C. C. et al. (2016). Recommender systems. Springer. Ahn, H. J. (2008). A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem. Information Sciences, 178, 37–51.

[2] Bennett, J., Lanning, S. & et al. (2007). The netflix prize. In Proceedings of KDD cup and workshop (Vol. 2007, p. 35). New York, NY, USA. Billsus, D., & Pazzani, M.J. (1998). Learning collaborative information filters. In Icml (Vol. 98, pp. 46–54).

[3] Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. The Journal of Machine Learning Research, 3, 993–1022.

[4] Bobadilla, J., Ortega, F., Hernando, A., & Bernal, J. (2012). A collaborative filtering approach to mitigate the new user cold start problem. Knowledge-Based Systems, 26, 225–238.

About the Author

Mouneswari Pentakota is currently pursuing her masters of Technology in the stream of Data Science and Jawaharlal Nehru and Technological University Guarajada, Vizianagaram. Her fascination is towards problem solving, innovate and research in the field of Data Science. Her works are related as Predictive models to extracts the data the business needs and help analyze the data.